Level 1A
Software Development Document

for the

Sounding of the Atmosphere using Broadband Emission Radiometry (SABER)

(Draft)

Submitted By:	______________________________	____________
	G.M.Beaver, GATS Instrument Performance Engineer	Date
Concurred By:	______________________________	____________
	L.L. Gordley, GATS CEO	Date

G & A Technical Software
Hampton, Virginia

Contract No: NAS1-20472
GATS DOC No: GATS-SABER-L1ASDD-98-1.V1

CHANGE INFORMATION PAGE

<table>
<tr><th colspan="4">List of Effective Pages</th></tr>
<tr><th colspan="2">Page Number</th><th colspan="2">Issue</th></tr>
<tr><td colspan="2">Title</td><td colspan="2">Original</td></tr>
<tr><td colspan="2"></td><td colspan="2">Original</td></tr>
<tr><td colspan="2"></td><td colspan="2">Original</td></tr>
<tr><td colspan="2"></td><td colspan="2">Original</td></tr>
<tr><td colspan="2"></td><td colspan="2">Original</td></tr>
<tr><td colspan="2"></td><td colspan="2"></td></tr>
<tr><th colspan="4">Document History</th></tr>
<tr><th>Document Number</th><th>Status/Issue</th><th>Publication Date</th><th>CCR Number</th></tr>
<tr><td></td><td>Original</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>

SABER Level 1A Software Development Document

1. Scope

This Software Design Document (SDD) describes an overall software design plan for SABER Level 1A processing software. The software sorts and merges unpacked data (Level 0B) into events, which are defined by scan modes of the SABER instrument. The main science data events are defined by individual scans (up or down) through the Earth's atmosphere. The data are converted from counts to engineering units. All data required to perform signal corrections and retrievals on an individual event will be merged into each scan event.

2 Referenced Documents

GIS
Level 0B format document GATS-SABER-L0BFFD-98.1.V1

3 System Overview

The SABER Level 1A data flow is shown in figure 1. The Payload Operation Center (POC) will pull one day's worth of raw telemetry (Level 0A) from the TIMED Mission Data Center (MDC). Level 0A consists of raw CCSDS packets concatenated together into one daily file. The POC will then decom and unpack the data and create Level 0B files, sorted by packet type (Instrument, Housekeeping, S/C) or data type (NMC, PVAT, Solar Geomagnetic). The Level 0B files are ASCII format and are described in the Level 0B Format Document.

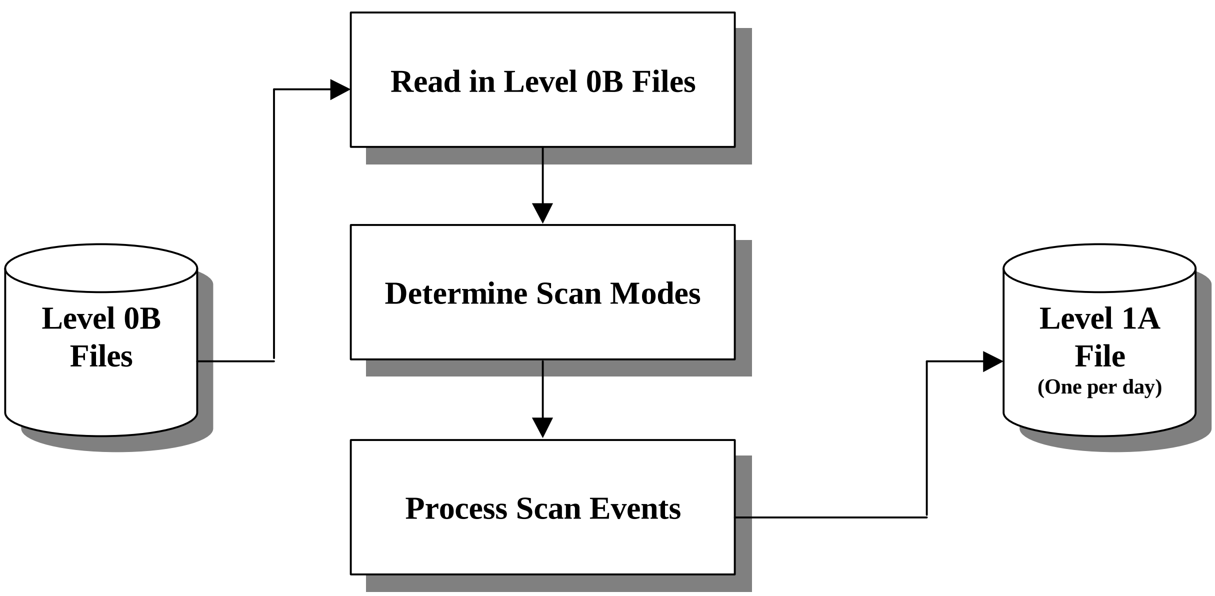

Figure 1: SABER Level 1A dataflow.

There are three types of Level 0B files: (1) those generated from CCSDS packets (*packet-generated*), (2) *ancillary* data and (3) status file containing record counts and error logs. The Level 0B files and their type are listed in Table 1. The packet-generated Level 0B are read into object arrays and converted to engineering units. Each Level 0B file has an associated C++ class, which has common header information containing date & time. The instrument data are processed to determine the event boundary times. Events are defined by reading in the scan mode flag and scan angle (from data collection class) and determining the beginning and ending boundary times of each defined scan event. Once the boundary times are determined, all modal objects are searched for records that fall between these times. Specific ancillary data needed to further process each event (such as NMC data, S/C attitude, and solar geomagnetic indices) are extracted from the ancillary data and loaded into an event header.

Level 0B File	Type	Format
Data Collection	Packet-Generated	ASCII
HouseKeeping	Packet-Generated	ASCII
NMC	Ancillary	NetCDF
PVAT	Ancillary	NetCDF
Solar Indices	Ancillary	NetCDF
Status (Record Count/Error Log)	Status	ASCII
Orbit Number	Ancillary	ASCII

Table 1: SABER Level 0B files

4 System Requirements

The Level 1A software must be able to run on SGI and Linux workstations. It must be able to read and write files across the Internet via NFS. Prior to calibration, the Level 1A package will be installed on the Space Dynamics Laboratory (SDL)'s SGI network so that calibration engineers will be able to access Level 1A files. During the actual instrument calibration, GATS engineers will generate Level 1A files for both groups. This will provide thorough testing for the operational Level 1A software.

5 Computer Software Configuration Items

The requirements for each CSCI discussed in Figure 1 are presented below.

5.1 CSCI: Read in Level 0B Files

This CSCI opens and reads in Level 0B files utilizing I/O methods contained within the modal classes created for Level 0B processing. Each modal class will be defined as an object array and all data for each modal file is read into the object array. For packet-generated data, each value is time tagged. For ancillary data (NMC, PVAT, Orbit Number and Solar Indices), the files are opened and accessible until the required location is determined. The Level 0B Status file contains record counts for each packet file as well as an error log. This file must be opened and read for error tracking and record count verification.

5.1.1 Requirements

The CSCI must open (for reading) each of the daily and previous day Level 0B files listed in Table 1. Level 0B files generated from packets must be read into time tagged records, based on the record count from the status file. The errors flagged in the status file are passed through via the quality flag defined for each record. The record structures are defined in the Level 1A format document. The ancillary Level 0B data need only be opened and accessible. The CSCI must be able to determine if a scan event began on the previous day, and include it in the current day processing.

5.1.2 Testing

Since the classes from Level 0B processing are reused, data can be printed out using the Level 0B output methods. The files generated by these methods can only contain data read in from the desired files, hence a point-by-point comparison can be performed between the Level 0B input files, and the data read in by the CSCI. Errors will also be implanted in Level 0B files to verify that errors are correctly detected and flagged. Test events will be generated which cross over day boundaries to ensure proper handling of events which started in the previous day.

5.2 CSCI: Determine Scan Modes

This CSCI must determine the scan mode as a function of time so that time boundaries for the beginning and ending of each scan mode event can be determined. The CSCI must read in the instrument scan mode flag, which is contained in the Level 0B main frame header (see Level 0B format document). Once a change in the scan mode flag is detected, a new event is declared and the event is processed. Since acquisition and adaptive scans can be further differentiated into up and down scans, the scan angle velocity must be checked to determine the scan direction. The possible scan modes are listed in Table 2.

Scan Mode	Scan Mode Flag	Scan Angle Velocity	Scan Angle Range [mrads]	Description
Adaptive Down	ADPTSCAN	TBD	TBD	Nominal scan down
Adaptive Up	ADPTSCAN	TBD	TBD	Nominal scan up
IFC	IFCBBXXX	TBD	TBD	Stare at Internal Flight Calibrator
Space Look	SPACLOOK	TBD	TBD	Stare at cold space
Acquisition Down	ACQNSCAN	TBD	TBD	Acquisition scan down
Acquisition Up	ACQNSCAN	TBD	TBD	Acquisition scan up
Lower Baffle Look	ACQNSCAN	TBD	TBD	Mirror Scans into lower baffle
Upper Baffle Look	ACQNSCAN	TBD	TBD	Mirror Scans into upper baffle
TBD 1-?	SPARE1-?	TBD	TBD	Spares for future modes

Table 2: SABER Scan Modes

5.2.1 Requirements

The CSCI must sort through the time-ordered data collection modal file, read the scan mode flag and scan mirror angle and determine beginning and ending time boundaries for each scan event identified in Table 2. The CSCI must trap errors in scan mode flag. The CSCI must also handle scan data that crosses day boundaries by terminating the event and saving it for the next day. Therefore, the current day and previous day data are required for input to correctly identify the first event for the current day in case it begins in the previous day.

5.2.2 Testing

During engineering calibration, the instrument will be commanded to go into each scan mode defined in Table 2. These data will be run through Level 1A processing. Demonstration and analysis can then verify that the designated scan modes are correct. Scan mode errors will be implanted in the level 0B files to verify that scan mode errors are successfully trapped. Events will be simulated that cross over into the next day to test such events that begin in one day and end in the next.

5.3 CSCI: Process Scan Events

Processing the scan events includes merging all required data for each designated scan event, determining the geometry as required, converting to Engineering Units, and writing out the data in the Level 1A format.

5.3.1 Requirements

The requirements for processing each scan event are dependent upon the scan mode. Each scan mode defined in Table 2 requires different ancillary data. The Level 1A file will be based on Adaptive Scan events, with every other scan staring at the IFC and/or cold space. The spare event modes are reserved for any on-orbit tests that may be required to characterize the instrument, or other scan modes developed after launch which can be commanded, and will have requirements derived as needed. The individual requirements for each scan mode are listed in Table 3. The final requirement for each processed scan event is to gather all data which is required to process each Adaptive Scan event, generate the mode-dependent event header and output it along with the data for that scan event. The format is defined in the SABER Level 1A format document.

5.3.2 Testing

Test data will be generated (during POC testing) by commanding the instrument to go into each possible scan mode. These data will be broken into scan events, and run through Level 1A processing. The output Level 1A file can be analyzed to verify that the data were correctly processed, merged and output in the correct format for each scan mode event. Implanted errors will also be tracked to ensure correct detection and flagging. These test data will also be merged with simulated SABER data using LIMS data for testing of SABER Level 1B processing.

Scan Mode	**Event Data Requirements**	**Processing Requirements**
Adaptive Down	Scan Mirror Angle Channel Voltages & Gains HouseKeeping NMC Profile for tangent point location. Solar Indices for current day. PVAT. Orbit Number. Bracketing IFC events. Bracketing Space-Look events Baffle-Looks (if scanned)	Convert to Engineering Units. Compute tangent point location. Extract NMC profile. Merge data.
Adaptive Up	Same as Adaptive Down	Same as Adaptive Down.
IFC	Scan Mirror Angle Channel Voltages & Gains IFC & Jones Source Temperatures	Convert to Engineering Units. Merge data.
Space Look	Scan Mirror Angle Channel Voltages & Gains PVAT.	Convert to Engineering Units. Compute tangent point location. Merge data.
Acquisition Down	Same as Adaptive Down	Same as Adaptive Down.
Acquisition Up	Same as Adaptive Up	Same as Adaptive Up.
Lower Baffle Look	Scan Mirror Angle Channel Voltages & Gains HouseKeeping	Convert to Engineering Units. Merge data.
Upper Baffle Look	Scan Mirror Angle Channel Voltages & Gains HouseKeeping	Convert to Engineering Units. Merge data.
Spare	Derived	Derived as needed.

Table 3: SABER Scan Mode Processing Requirements

6 Acronym List

APID	Application Identifier
APL	Applied Physics Laboratory
CCSDS	Consultative Committee for Space Data Systems
CSCI	Computer Software Configuration Item
CVT	Current Value Table
DLL	Dynamic Link Library
FTP	File Transfer Protocol
GATS	Gordley and Associates Technical Software
GSE	Ground Support Equipment
HALOE	HALogen Occultation Experiment
H, S & P	Health, Safety and Performance
H/W	Hardware
ICD	Interface Control Document
IDL	Interactive Development Language

IFC	In-Flight Calibrator
JHAPL	Johns Hopkins Applied Physics Laboratory
LIMS	Limb Infrared Monitoring of the Stratosphere
MDC	Mission Data Center
MOC	Mission Operation Center
POC	Payload Operations Center
PVAT	Position, Velocity, Attitude and Time
RT	Real Time
SABER	Sounding of the Atmosphere using Broadband Emission Radiometry
SAGE	Stratospheric Aerosol and Gas Experiment
S/C	Spacecraft
SDD	Software Development Document
SDL	Space Dynamics Laboratory
S/W	Software
TCP/IP	Transmission Control Protocol over Internet Protocol
TIMED	Thermosphere, Ionosphere, Mesosphere, Energetics, Dynamics
UTP	Unshielded Twisted Pair

7 Appendix A: Level 1A File Format Description

PURPOSE

The purpose of this appendix is to define the content and format of the SABER Level 1A file. This file will be a product derived from Level 0B modal files and will contain instrument, housekeeping, spacecraft (S/C) and ancillary data *in engineering units*. The Level 1A file will be separated into scan events that are determined by the scan mode and scan mirror position and velocity, and used as input to the Level 1B processing.

BACKGROUND

The Level 1A file will be the output from Level 1A processing, which reads in modal files, converts to engineering units, determines scan event boundary times, and merges data into scan events. For Level 1A processing this involves reading in the scan mode flag and scan angle and determining boundary times for beginning and ending angles of the type of scan. Scan events are listed in Table 1.

Event Definition	**Scan Mode Flag**	**Scan Angle Direction**	**Scan Angle Range [mrads]**	**Event Type String**	**Description**
Adaptive Down	ADPTSCAN	TBD	TBD	ADAPT_UP	Nominal scan down
Adaptive Up	ADPTSCAN	TBD	TBD	ADAPT_DN	Nominal scan up
IFC	IFCBBXXX	TBD	TBD	IFCBBXXX	Stare at Internal Flight Calibrator
Space Look	SPACLOOK	TBD	TBD	SPACLOOK	Stare at cold space
Acquisition Down	ACQNSCAN	TBD	TBD	ACQ_UP	Acquisition scan down
Acquisition Up	ACQNSCAN	TBD	TBD	ACQ_DN	Acquisition scan up
Lower Baffle Look	ACQNSCAN	TBD	TBD	LBAFF	Mirror Scans into lower baffle
Upper Baffle Look	ACQNSCAN	TBD	TBD	UBAFF	Mirror Scans into upper baffle
Fast Scan Down	FASTSCAN	TBD	TBD	FAST_DN	Scan down at fast rate
Fast Scan Up	FASTSCAN	TBD	TBD	FAST_UP	Scan up at fast rate
TBD 1-?	SPARE1-?	TBD	TBD	TBD	Spares for future modes

Table 1: SABER Scan Modes

REQUIREMENTS
The Level 1A file will contain data merged from Level 0B files with samples taken between beginning and ending times of each defined event, determined by the scan mode. The scan events are determined by the scan mode flag and the position and velocity of the scan mirror. Each record will be tagged with a time and scan mode flag (see Table 1). The record time will be in msec since UT midnight, and will be the time when the first measurement field was sampled.

IMPLEMENTATION
The Level 1A file will be divided into adaptive scan events containing instrument data sampled between scan event time boundaries, along with ancillary data required to process each event. Each adaptive scan event will have a scan event header, followed by data that is scan mode event dependent. The event headers contain time information and ancillary data required for processing the event.

FORMAT
The Level 1A format will be ASCII with free format records and fields, (easily represented by a structure). ASCII format ensures cross-platform compatibility, since Linux, SGI and NT workstations will need to share SABER data. Each file contains one day of data. There will be 3 record types; (1) Instrument data, sampled at 22.7 Hz, (2) HouseKeeping data, sampled at 0.063 Hz, and (3) S/C data, sampled at 2 Hz. Based on these sampling rates, the Level 1A file will contain roughly 1 housekeeping record (type 2) for every 32 S/C records (type 3) for every 360 Instrument data records (type 1). The first 4 fields in each record (regardless of type) will be header fields consisting of time, record type, status, and number of fields remaining. The rest of the fields are record-dependent, which are described below.

SCAN MODE EVENT HEADER DEFINITION
The Level 1A file is based on adaptive scan mode events. Each time the SABER mirror makes a scan through the Earth's atmosphere, a scan event is defined. The Level 1A file is thus a collection of scan events. For each scan event, the bracketing IFC and Space-look data are included, as well as any baffle-look data if taken. Exactly what data is contained in each event appears in the scan event header. The scan event header will contain the following fields: Event Number (for current day), Year, Date, Orbit Number, Begin Time, End Time, Orbit Number, Data Type, Number of records, where

Event Number is the scan number for the current day,
Year is the 4-digit year,
Date is the 3-digit day-of–year (001-365),
Orbit Number is the TIMED orbit number,
Data Type: Number of Records describes the data which follow in this event and is repeated for all the merged data in the current event and is defined in Table 2.

The format for each data type is described below. Note that each event type has a self-described format, since a reader routine can tell from the (Data Type: Number of Records) fields in the event header precisely what data follows.

Data Type	Description
DC	Instrument Data
HK	HouseKeeping Data
NMC	Pressure Temperature Altitude from NMC data
PVAT	Position, Velocity, Attitude and Time data
SOL	Solar Geomagnetic Indices

Table 2: Data types which appear in SABER Level 1A file.

RECORD TYPE DEFINITIONS

The 6 types of records written in a Level 1A file are described below.

DC Record Type

This record type contains data from the Data Collection packets. "PGA" refers to the programmable Gain Amplifier setting in engineering units. Each channel has 3 gain range settings, described in the SABER Instrument Specification document.

Field	Value	Units
1	Record Start Time	Milliseconds since UT midnight
2	Scan Angle	milliradians
3-22	Channel 1-10 PGA Setting	[unitless]
23-32	Channel 1-10 Voltage	[Volts]
33	Quality Flag	N/A

Table 3: Level 1A DC record type definition.

The structure which defines record type 1 follows:

```
// Instrument Data Record
typedef struct {
      long int startTime;       // Record start time (msec since UT midnight
      double scanAngle;         // Scan Mirror Angle [mrads]
      double gain[10];          // Channel 1-10 PGA settings []
      int chan[10];             // Channel 1-10 output voltages [counts]
      double qualityFlag;       // Record quality indicator [tbd]
   } DC_Record;
```

HK Record Type

Contains housekeeping data subcommed from the Data Collection packets, as well as any data from housekeeping mode packets.

Field #	Mnemonic	Value	Units
1		Record Start Time	Milliseconds
HouseKeeping Temperature Monitors			
5	tfo1vg1	Temperature 1 Focal Plane 1	(voltage)
6	tfo1vg2	Temperature 1 Focal Plane 2	(voltage)
7	tfo1c	Temperature 1 Focal Plane	(current)
8	tfo2vg1	Temperature 2 Focal Plane 1	(voltage)
9	tfo2vg2	Temperature 2 Focal Plane 2	(voltage)
10	tfo2c	Temperature 2 Focal Plane	(current)
11	tfo3v	Top of Cold Link #3	(voltage)
12	tfo3c	Top of Cold Link #3	(current)
13	tfo4v	Top of Cold Link #4	(voltage)
14	tfo4c	Top of Cold Link #4	(current)
15	tfo5v	Bottom of Cold Link #5	(voltage)
16	tfo5c	Bottom of Cold Link #5	(current)
17	tfo6v	Bottom of Cold Link #6	(voltage)
18	tfo6c	Bottom of Cold Link #6	(current)
19	tto1v	Radiator Near Aperture Top	
20	tto2v	Radiator Near Aperture Bottom	
21	tto3v	Baffle Near Aperture Top	
22	tto4v	Baffle Hot Spot	

23	tto5v	Baffle Near Rear Joint	
24	tto6v	Wall Near Bearing Right	
25	tto7v	Wall Near Encoder	
26	tto8v	Encoder Mount Front	
27	tto9v	Encoder Mount Back	
28	tto10v	Fore-optics S/C	
29	tto11v	IFC #1	(voltage)
30	tto11c	IFC #1	(current)
31	tto12v	IFC #2	(voltage)
32	tto12c	IFC #2	(current)
33	tto13v	IFC #3	(voltage)
34	tto13c	IFC #3	(current)
35	tto14v	Fore-optics Radiator	
36	tto15v	Chopper Base Right	
37	tto16v	Chopper Base Left	
38	tm01v	Refrigerator Mount Top	
39	tm02v	Refrigerator Mount Bottom	
40	tm03v	Radiator at Ref Mount Base	
41	tm04v	Radiator at Electronics Box	
42	tm05v	Electronics Box Hot Spot	
43	tm06v	Electronics Box Back	
44	tm07v	Radiator at RFE Box	
45	tm08v	RFE Box Hot Spot	
46	tm09v	RFE Box Back	
47	tco1v	Cover Deploy System	
48	tref	Reference Voltage for Current	
HouseKeeping Power Monitors			
49	v1p5	DC/DC#1 +5V, C&DH	Volts
50	v2p15	DC/DC#2 +15V, Housekeeping	
51	v2m15	DC/DC#3 -15V, Housekeeping	
52	v4p5	DC/DC#4 +5V, Scan, Analog Count	
53	v5p15	DC/DC#5 +15V, Scan, Analog Count	
54	v5m15	DC/DC#6 -15V, Scan, Analog Count	
55	v6p15	DC/DC#7 +15V, Scan Drive	
56	v7p15	DC/DC#8 +15V, Signal Process	
57	v7m15	DC/DC#9 -15V, Signal Process	
58	v9p5	DC/DC#10 +5V, Signal Process	
59	v10p28	DC/DC#11 +28V, Refrigerator	
HouseKeeping Calibration Sources			
60	bbsetv	Blackbody Set Voltage	
61-63	jscur[3]	Jones Source Currents	
HouseKeeping Refrigerator			
64	rcs	Refrigerator Compressor Position	
65	rbs	Refrigerator Balancer Position	
66	rcc	Refrigerator Compressor Current	mAmps
67	rbc	Refrigerator Balancer Current	mAmps
68	rao	Refrigerator Accelerometer Out	
69	trch	Temperature Cold Head	
HouseKeeping Status			
70	stat1Hbeat	Heartbeat, Error Counters	

71	stat2Uplink	Uplink Status	
72	stat3Onoff	On/Off Status	
73	stat4Lnktmp	TFO3V, Ref Cold Link Temp	
74	stat5Intrpt	Interrupt, Subsystem Status	
		HouseKeeping Register Values	
75	ti	Time Interrupt Compare Word	
76-85	ge[10]	Auto Gains (on/off)	
86-95	ghtp[10]	Auto Gain High Trip Points	
96-105	gltp[10]	Auto Gain Low Trip Points	
106-115	o[10]	Channel Offsets	Volts
116	mdump	EEPROM Dump Block Start Position	
117-119	jssetcur[3]	Jones Source Set Currents	
120	bbt	Blackbody Set Temperature	
121	adscoff1	Adaptive Scan Offset 1	
122	adscml1	Adaptive Scan Mirror Limit 1	
123	adscoff2	Adaptive Scan Offset 2	
124	adscml2	Adaptive Scan Mirror Limit 2	
125-127	dfsa[3]	Data Formatter Start Addresses	
128	rcon1	Refrigerator Control Word 1	
129	rcon2	Refrigerator Control Word 2	
130	crccomp	CRC Compare Value	
131	crccur	CRC Present Value	
132	retupad	Return From Uplink Address	
133	astemp	Adaptive Scan Temporary Store	
134	ad1_2p	Adaptive Scan 1/2 Peak Value	
135	ml1_2p	Mirror Location at 1/2 Peak	
136	mfsp1	Mirror Full Sweep Position 1	
137	mfsp2	Mirror Full Sweep Position 2	

The structure(s) that defines record type 2 follows;

```
// HouseKeeping Record
typedef struct {
   long int startTime;       // Record start time (msec since UT midnight
   HK_Temp_Mon  temp_mon;    // Temperature Monitors Structure (ch 0-43)
   HK_Power_Mon power_mon;   // Power Monitors Structure (ch 44-54)
   HK_Cal_Src   cal_src;     // Calibration Sources Structure (ch 55-58)
   HK_Refrig    refrig;      // Refrigerator Structure (ch 59-64)
   HK_Status    hkStatus;    // Status Structure (ch 65-69)
   HK_Reg_Val   reg_val;     // Register Values Structure (ch 70-132)
} HkRec;

   // HouseKeeping Temperature Monitors
   typedef struct {
      double       tfo1vg1;     // Temperature 1 Focal Plane 1 (voltage)
      double       tfo1vg2;     // Temperature 1 Focal Plane 2 (voltage)
      double       tfo1c;       // Temperature 1 Focal Plane (current)
      double       tfo2vg1;     // Temperature 2 Focal Plane 1 (voltage)
      double       tfo2vg2;     // Temperature 2 Focal Plane 2 (voltage)
      double       tfo2c;       // Temperature 2 Focal Plane (current)
      double       tfo3v;       // Top of Cold Link #3 (voltage)
      double       tfo3c;       // Top of Cold Link #3 (current)
      double       tfo4v;       // Top of Cold Link #4 (voltage)
      double       tfo4c;       // Top of Cold Link #4 (current)
      double       tfo5v;       // Bottom of Cold Link #5 (voltage)
      double       tfo5c;       // Bottom of Cold Link #5 (current)
      double       tfo6v;       // Bottom of Cold Link #6 (voltage)
      double       tfo6c;       // Bottom of Cold Link #6 (current)
      double       tto1v;       // Radiator Near Aperture Top
      double       tto2v;       // Radiator Near Aperture Bottom
      double       tto3v;       // Baffle Near Aperture Top
      double       tto4v;       // Baffle Hot Spot
      double       tto5v;       // Baffle Near Rear Joint
```

```
   double       tto6v;       // Wall Near Bearing Right
   double       tto7v;       // Wall Near Encoder
   double       tto8v;       // Encoder Mount Front
   double       tto9v;       // Encoder Mount Back
   double       tto10v;      // Fore-optics S/C
   double       tto11v;      // IFC #1 (voltage)
   double       tto11c;      // IFC #1 (current)
   double       tto12v;      // IFC #2 (voltage)
   double       tto12c;      // IFC #2 (current)
   double       tto13v;      // IFC #3 (voltage)
   double       tto13c;      // IFC #3 (current)
   double       tto14v;      // Fore-optics Radiator
   double       tto15v;      // Chopper Base Right
   double       tto16v;      // Chopper Base Left
   double       tm01v;       // Refrigerator Mount Top
   double       tm02v;       // Refrigerator Mount Bottom
   double       tm03v;       // Radiator at Ref Mount Base
   double       tm04v;       // Radiator at Electronics Box
   double       tm05v;       // Electronics Box Hot Spot
   double       tm06v;       // Electronics Box Back
   double       tm07v;       // Radiator at RFE Box
   double       tm08v;       // RFE Box Hot Spot
   double       tm09v;       // RFE Box Back
   double       tco1v;       // Cover Deploy System
   double       tref;        // Reference Voltage for Current
} HkTempMon;

// HouseKeeping Power Monitors
typedef struct {
   double       v1p5;        // DC/DC#1 +5V, C&DH
   double       v2p15;       // DC/DC#2 +15V, Housekeeping
   double       v2m15;       // DC/DC#3 -15V, Housekeeping
   double       v4p5;        // DC/DC#4 +5V, Scan, Analog Count
   double       v5p15;       // DC/DC#5 +15V, Scan, Analog Count
   double       v5m15;       // DC/DC#6 -15V, Scan, Analog Count
   double       v6p15;       // DC/DC#7 +15V, Scan Drive
   double       v7p15;       // DC/DC#8 +15V, Signal Process
   double       v7m15;       // DC/DC#9 -15V, Signal Process
   double       v9p5;        // DC/DC#10 +5V, Signal Process
   double       v10p28;      // DC/DC#11 +28V, Refrigerator
} HkPowerMon;

// HouseKeeping Calibration Sources
typedef struct {
   double       bbsetv;      // Blackbody Set Voltage
   double       jscur[3];    // Jones Source Currents
} HkCalSrc;

// HouseKeeping Refrigerator
typedef struct {
   double       rcs;         // Refrigerator Compressor Position
   double       rbs;         // Refrigerator Balancer Position
   double       rcc;         // Refrigerator Compressor Current
   double       rbc;         // Refrigerator Balancer Current
   double       rao;         // Refrigerator Accelerometer Out
   double       trch;        // Temperature Cold Head
} HkRefrig;

// HouseKeeping Status
typedef struct {
   short int    stat1Hbeat;  // Heartbeat, Error Counters
   short int    stat2Uplink; // Uplink Status
   short int    stat3Onoff;  // On/Off Status
   double       stat4Lnktmp; // TFO3V, Ref Cold Link Temp
   short int    stat5Intrpt; // Interrupt, Subsystem Status
} HkStatus;

// HouseKeeping Register Values
typedef struct {
   short int    ti;          // Time Interrupt Compare Word
   char         ge[10];      // Auto Gains (on/off)
   double       ghtp[10];    // Auto Gain High Trip Points
   double       gltp[10];    // Auto Gain Low Trip Points
   double       o[10];       // Channel Offsets
   short int    mdump;       // EEPROM Dump Block Start Position
   double       jssetcur[3]; // Jones Source Set Currents
   double       bbt;         // Blackbody Set Temperature
   double       adscoff1;    // Adaptive Scan Offset 1
   double       adscml1;     // Adaptive Scan Mirror Limit 1
   double       adscoff2;    // Adaptive Scan Offset 2
   double       adscml2;     // Adaptive Scan Mirror Limit 2
```

```
      short int    dfsa[3];      // Data Formatter Start Addresses
      short int    rcon1;        // Refrigerator Control Word 1
      short int    rcon2;        // Refrigerator Control Word 2
      short int    crccomp;      // CRC Compare Value
      short int    crccur;       // CRC Present Value
      short int    retupad;      // Return From Uplink Address
      short int    astemp;       // Adaptive Scan Temporary Store
      short int    ad1_2p;       // Adaptive Scan 1/2 Peak Value
      short int    ml1_2p;       // Mirror Location at 1/2 Peak
      short int    mfsp1;        // Mirror Full Sweep Position 1
      short int    mfsp2;        // Mirror Full Sweep Position 2
   } HkRegVal;
```

PVAT Record Type Definition

This record type contains data from Pointing Velocity Attitude Time (PVAT) file.

Field	Value	Units
1	Record Start Time	Milliseconds since UT midnight
2	TBD	TBD

NMC Record Type Definition

This record type contains pressure, altitude and temperature data for the current SABER scan. The profile comes from the NMC NetCDF file. The profile selected is based on 60km tangent point location and time for the unrefracted ray.

Field	Value	Units
1	Profile Latitude (60 km TP)	Degrees (0=equator, -90=S. Pole, +90=North Pole)
2	Profile Longitude (60 km TP)	Degrees E
3	TIMED Orbit Number	
4	SABER Scan #	
5	Date	YYYYDDD
6	Time (60 km)	Msec since midnight

The structure which defines record type 1 follows:

```
// Spacecraft Data Record
typedef struct {
      long int startTime;       // Record start time (msec since UT midnight
} HC_Record;
```

SOL Record Type Definition

TBD